AFRL-SR-AR-TR-04-

0657

REPORT DOCUMENTATION PAGE

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE** Final	**3. DATES COVERED** *(From - To)* 1 June 2001 - 31 May 2004

4. TITLE AND SUBTITLE Cooperative Mission Control for Unmanned Air Vehicles	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER F49620-01-1-0348
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) David A. Castanon Christos G. Cassandras	**5d. PROJECT NUMBER**
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Boston University Dept. Elect. & Comp. Engineering Boston, MA 02215	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research 4015 Wilson Blvd Mail Room 713 Arlington, VA 22203	**10. SPONSOR/MONITOR'S ACRONYM(S)** AFOSR
NM	**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution Statement A. Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report documents the results of our investigations towards a theory of cooperative mission control for groups of unmanned air vehicles. The investigations were focused on two levels of control: (1) Strategy Development, to determine missions of interest and corresponding strategies for accomplishing those missions in centralized and distributed ssettings, and (2) Dynamic Vehicle Assignment and Scheduling, which determines activities, routes and schedules for individual vehicles to accomplish desired missions of interest and strategies.

20050104 035

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
U	U	U	UU		David A. Castanon
					19b. TELEPHONE NUMBER *(Include area code)*

Cooperative Mission Control for Unmanned Air Vehicles

Final Report
Grant No. F49620-01-1-0348

Submitted to
Air Force Office of Scientific Research
Mathematics and Space Science Division
4015 Wilson Blvd, Room 713
Arlington, VA 2203-1954

Principal Investigator
David A. Castañón
Dept. Elect. and Comp. Engineering
Boston University
Boston, MA 02215 USA
dac@bu.edu

Co-Principal Investigator
Christos G. Cassandras
Dept. Manufacturing Engineering
Boston University
Boston, MA 02215 USA
cgc@bu.edu

September, 2004

1 Introduction and Overview

1.1 Introduction

In Joint Vision 2010 [1], the Chairman of the Joint Chiefs of Staff outlined a vision of effective, efficient armed services for the next century. This document stressed the importance of "information superiority" which he defined as "The capability to collect, process, and disseminate an uninterrupted flow of information while exploiting or denying and adversary's ability to do the same." Enhanced command, control, and intelligence provided by information superiority will transform the traditional functions of maneuver, strike, protection and logistics into a new conceptual framework for operations. The document identified the tactical use of autonomous assets as a key technology that will enable the commanders to conduct dominant maneuvers and precision engagements.

Since the issuance of [1], the United States has been involved in military conflicts in Afghanistan and Iraq, that saw increased use of autonomous tactical aircraft such as Predators and Global Hawks for surveillance and target prosecution. In addition, the U. S. Air Force has proceeded with the development of Unmanned Combat Air Vehicles (UAVs) in order to accomplish missions that may be difficult for manned aircraft due to survivability or other reasons. These missions include Suppression of Enemy Air Defenses (SEAD), moving target attack, fixed target attack, Intelligence Surveillance and Reconnaissance (ISR), jamming, theater missile defense, and counter weapons of mass destruction. UAV prototypes have demonstrated so far the capability for unmanned flight. Other services such as the U. S. Army and U. S. Navy have been actively pursuing the development of unmanned combat air, ground and underwater vehicles.

One of the major limitations of current tactical unmanned air vehicles (UAVs) is the high level of human control required to operate a single vehicle. Currently, aircraft such as Predators require two or more operators per aircraft to provide information processing, navigation and mission control decisions; this places limits on the use of large numbers of autonomous assets acting in a coordinated manner; the use of unmanned air vehicles in Afghanistan and Iraq consisted primarily of individual missions involving single aircraft. The new range of UAVs will have the capability of autonomous sensing and control, and will be capable of **cooperatively** detecting, locating, classifying and prosecuting intelligent, mobile targets.

In order to achieve the full potential of unmanned tactical vehicles, many of the decisions made by human operators will be generated by an intelligent cooperative control system, guided by objectives and constraints provided by human operators. This cooperative control will have to solve complex dynamic decision problems associated with mission planning and control, in an unstructured and uncertain environment, in near real time, and without human intervention [17, 18, 19], adapting in an intelligent fashion to events which arise in a hostile, uncertain and rapidly evolving environment. This requires new technology for *cooperative mission control*, capable of selecting and coordinating the activities of multiple heterogeneous platforms to achieve a common objective.

This report documents the results of our investigations towards a theory of cooperative mission control for groups of unmanned air vehicles. The investigations were focused on two levels of control: (1) Strategy Development, to determine missions of interest and corresponding strategies for accomplishing those missions in centralized and distributed settings, and (2) Dynamic Vehicle Assignment and Scheduling, which determines activities, routes and schedules for individual vehicles to accomplish desired missions of interest and strategies. The main results of our research were:

- A new theory for developing strategies for target prosecution among heterogeneous vehicles that use unreliable resources [5, 10]. The results are based on a stochastic dynamic programming approach, and use a constraint relaxation approach to generate approximate solutions that are made feasible through a model-predictive control approach. These results were extended to include dynamic sensor management near the end of the program.
- New asynchronous distributed algorithms for modifying task assignments among members of UAV teams in response to dynamic scenarios where tasks arrive and depart, when information among team members can differ due to communication delays [9]. Our results in [9] establish that the asynchronous algorithms have proven stability properties, such as guaranteed convergence to optimal assignments once scenarios stabilize.
- A receding horizon control for simultaneous path planning and task assignment [6, 8] that is related to potential function approaches to path planning [7]. Assignment occurs implicitly as vehicles are attracted to tasks, and avoid overlap with other vehicles. The inputs to this algorithm are the specific tasks associated with the strategies selected in [10].

The above results were provided to industry researchers for use in experimental cooperative control programs such as DARPA's MICA and AFRL DACC and DCAT programs. These collaborations

enabled us to evaluate the relevance of the results without incurring the commensurate cost in developing and using costly simulations. These industry researchers are actively involved in developing mission control software for programs such as J-UCAS, and provide a transition path for our results.

The results reported below include the works of the two principal investigators, Prof. D. Castañón and Prof. C. Cassandras and their three Ph. D. students, Mr. Wei Li, Mr. Xiang Ma and Major Darryl Ahner, US Army. The three students are scheduled to complete their dissertations in 2005. The following sections provide an overview of the key results of the research.

2 Strategy Development

2.1 Formulation

An important focus of the research was dynamic of UAV activities in the presence of uncertainty. In particular, we were interested in models where the probability distribution of future uncertain events depends explicitly on the current choice of actions. Such models capture effects such as the risk incurred by UAVs in selecting alternate paths, or the uncertain effects of actions such as attacking targets where the outcome is uncertain.

Our initial investigations focused on abstractions of dynamic scheduling problems for a group of autonomous vehicles carrying nonrenewable resources (e.g. munitions) that must be assigned to tasks which arrive in a dynamic, uncertain fashion. Furthermore, resources may fail to successfully complete a task. The mission control problem in this context is to schedule resources over time to assign to individual tasks based on vehicle capabilities, available resources and available information, so as to maximize the expected task completion value. We assume in this framework that vehicles are able to observe resource failures during each time period through mechanisms such as battle damage assessment. Thus, one can exploit this dynamic information and adapt future decisions based on the observation of success or failure events.

The problem we focused on was a stochastic control problem with multiple stages. The simplest two-stage version of the problem is summarized as follows: Assume that there are N tasks, indexed by $i = 1, \ldots, N$, and that there are a total of M non-renewable homogeneous resources which can be assigned to each task over two possible stages. Each task has a numerical value V_i which is obtained by completing the task. Each resource has a cost C of assigning the resource. When a resource is assigned to task i in stage k, the event that the resource completes the task successfully has probability $p_i(k)$, and this event is independent of events generated by other resource assignments.

Let $x_i(1)$ denote the number of resources assigned to task i at stage 1. The probability that task i is not completed by these resource assignments is obtained from the above independence assumptions as:

$$P_S(i, 1) = (1 - p_i(1))^{x_i(k)} \tag{1}$$

At the end of stage 1, the set of completed tasks will be observed. This set will denote the task

state vector, and is a Boolean vector $\underline{\omega} \in \Omega = \{0,1\}^N$; $\omega_i = 0$ denotes that task i was completed in stage 1, and $\omega_i = 1$ denotes the complementary event for task i. Given a vector of stage 1 resource allocations $\underline{x}(1)$, equation (1) induces a probability distribution $P(\underline{\omega}|\underline{x}(1))$ on the possible outcomes. The stage 2 allocations are strategies which depend on the specific outcome $\underline{\omega}$ which is observed, as $\underline{x}(2,\underline{\omega})$.

The evolution of the task state $\underline{\omega}$ across stages is a Markov process with states in Ω. Given resource allocations $\underline{x}(1)$ and recourse strategies $\underline{x}(2,\underline{\omega})$, the probability that task i is not completed either in stage 1 or in stage 2 is given by

$$P_S(i,2) = \sum_{\underline{\omega} \in \Omega} P(\underline{\omega}|\underline{x}(1)) I(\omega_i = 1)(1-p_i(2))^{x_i(2,\underline{\omega})} \tag{2}$$

where $I(\cdot)$ is the indicator function, and

$$P(\underline{\omega}|\underline{x}(1)) = \prod_{\{i|\omega_i=0\}} [1-(1-p_i(1))^{x_i(1)}] \prod_{\{j|\omega_j=1\}} (1-p_j(1))^{x_j(1)} \tag{3}$$

The optimal control problem is to select resource allocations $\underline{x}(1)$ and recourse strategies $\underline{x}(2,\underline{\omega})$ to minimize the expected incomplete task value plus the expected cost of using resources:

$$E\{\sum_{i=1}^{N} V_i P_S(i,2) + C \sum_{i=1}^{N} (x_i(1) + x_i(2,\underline{\omega})\} \tag{4}$$

subject to the constraints

$$\sum_{i=1}^{N} (x_i(1) + x_i(2,\underline{\omega})) \leq M \quad \text{for all } \underline{\omega} \in \Omega \tag{5}$$

$$x_i(1) \in \{0,1,\ldots,M\}, x_i(2,\omega) \in \{0,1,\ldots,M\} \quad \text{for all } \omega \in \Omega, i \in \{1,\ldots,n\} \tag{6}$$

The above problem is a two-stage stochastic control problem with a discrete state space that grows exponentially in the number of tasks N, and a discrete decision space which also grows exponentially in the number of tasks.

2.2 Model Predictive Control Algorithms

In our papers [5, 10], we developed the stochastic dynamic programming solution to the above problem. This solution is impractical for real-time mission control algorithms. As an alternative, we developed a new formulation based on techniques from model predictive control (MPC). Instead

of using the model in (1)-(6), we solve a *relaxed* version of the problem obtained by replacing the 2^N constraints in eq.(5) by one average resource utilization constraint:

$$\sum_{i=1}^{N} \sum_{\underline{\omega} \in \Omega} P(\omega | \underline{x}(1))(x_i(1) + x_i(2, \underline{\omega}) \leq M \tag{7}$$

The new optimization problem used for the MPC approach is to minimize (4) subject to constraints in (7,6). The solution of this optimization problem determines $\underline{x}(1)$, the first stage resource allocations, as well as strategies for future allocations. Only the first stage allocations are implemented; subsequently, based on the observed outcome $\underline{\omega}$, the second stage allocations are determined using a static network optimization problem.

Note that every feasible strategy for the constraints in (5) also satisfies (7). Thus, the MPC problem is a relaxation of the original problem, expanding the set of admissible strategies, and its solution is an optimistic lower bound on the optimal performance in the original problem.

The main theoretical result in [5, 10] is an equivalence result that establishes the existence of optimal strategies that use only "local" feedback information. That is, the resource allocations in stage 2 for task i, denoted as $x_i(2, \underline{\omega})$), can be restricted to depend only on the state of task i, ω_i. This allows the use of Lagrange duality to achieve a separable solution among single task problems, coordinated through dual prices. The result is an $O(N \log N)$ algorithm for exact solution of the two-stage relaxed integer stochastic dynamic programming problem, for N tasks. This fast algorithm yields approximate first stage decisions in the MPC approach. Once the outcome of the first stage actions are observed, the second stage decisions are re-optimized using another fast algorithm.

2.3 Numerical Experiments

In order to evaluate the effectiveness of the proposed MPC algorithms, we conducted several experiments comparing the following algorithms:

1. The exact stochastic dynamic programming (SDP) solution, obtained by enumerating the possible first stage allocations and finding a global minimum.

2. An approximate exponential complexity algorithm (IA) described in [5, 10].

Tasks	Resources	IA Alg.	MPC Alg.	Worst MPC
7	7	100%	99.92%	98.6%
7	9	100%	99.82%	99.18%
7	11	100%	99.996%	99.86%
9	7	100%	99.91%	98.30%
9	9	100%	99.89%	99.48%
9	11	100%	99.82%	98.96%
11	7	100%	99.92%	99.53%
11	9	100%	99.93%	99.56%
11	11	100%	99.74%	99.19%

Table 1: Performance of MPC and IA algorithms as percent of value completed by SDP

3. The MPC algorithm developed in [5, 10] discussed above.

The first set of experiments consisted of random problems with 7 to 11 tasks, with task values selected randomly in the range of 1-10, and task success probabilities selected randomly in the interval $[0.7, 0.9]$. The number of resources varied from 7 to 11. For each experiment, we generated 100 random problems, and obtained the solutions (in terms of value achieved) given by the SDP, IA and MPC algorithms. We computed the performance of the two suboptimal algorithms as percentage of the value achieved by the optimal SDP algorithm, averaged over the 100 problems. We also computed the worst case percentage difference in performance between the MPC algorithm and the SDP algorithm. The results are summarized in Table I.

The results in Table I establish that the MPC algorithm yields near-optimal performance: The worst case performance across 900 problems tested was within 2% of the optimal SDP performance, and the average performance was within 0.3% of the optimal SDP performance.

The second set of experiments used problems with 16 and 20 tasks, and with a varying number of resources from 12 to 20. For these problems, computing the exact dynamic programming solution using enumerative techniques was prohibitively long. As a reference point, it required 3 days on a LINUX Pentium 1.7 GHz workstation to solve 100 instances of the 11 task problem. We compared results only for IA and MPC algorithms. The statistics reported are the percentage of the IA value that is achieved by the MPC algorithm. We report both the average percentage across 100 problems and the worst-case percentage over the 100 problems for each experiment. The results are summarized in Table II.

Tasks	Resources	Ave. MPC	Worst MPC
16	12	99.81%	99.22%
16	16	99.82%	99.33%
16	20	99.92%	99.67%
20	12	99.85%	99.46%
20	16	99.85%	99.52%
20	20	99.88%	99.37%

Table 2: Performance of MPC algorithm as percent of value completed by IA algorithm

The results in Table II confirm the near optimal behavior of the Model Predictive Control algorithm. The average performance is within 0.2% of the performance of the IA algorithm, and the worst case performance is within 1% of the performance of the IA algorithm. In terms of computation time, the IA algorithm requires over 13 minutes to solve a single instance of a 20 task, 20 resource problem on a Pentium 1.4 GHz workstation running Linux. In contrast, the MPC algorithm solved 100 instances of 1000 task, 1000 resource problems in a total of 3.5 seconds. This is nearly five orders of magnitude faster, making it suitable for real-time mission control.

2.4 Extensions

The above results have been extended in several significant directions, as discussed in [5, 10]. The first extension is inclusion of multiple period dynamics, extending the two-period results above. The second extension is to allow for new target arrivals in future periods. A third extension is to consider targets with unknown values, that require the use of sensors to confirm their true value. This extension corresponds to targets with uncertain identity, that must be identified in order to determine their true value. A fourth extension was to have different classes of non-renewable resources, corresponding to different types of munitions.

For each of the above extensions, we developed fast MPC algorithms, and evaluated the performance of these algorithms on simulated problems. An experiment documented in [5, 10] used two different types of munitions. Each type j had different effectiveness p_{ij} for each task i. Task values were generated randomly in the range of 1 to 10, with random values for p_{ij} in the range of 0.7 to 0.9. We compared the MPC performance after averaging over 10,000 Monte Carlo trials to the optimistic bound obtained by the solution of the approximate model used by the MPC controller, for 100 random instances each of 10 task, 50 task and 100 task problems. The number of available

Tasks	Res. I	Res. II	Ave. MPC	Worst MPC
10	5	5	98.5%	95.8%
50	25	25	99.5%	99.2%
100	50	50	99.7%	97.9%

Table 3: Performance of MPC algorithm as percentage of task value completed by Mixed Strategy upper bound for two resource types

resources in each problem was set so that the total number of resources equaled the number of tasks, and there were equal numbers of two types of resources.

The results of this experiment, summarized in Table III, show that the MPC controller achieves expected performance is over 95% of an optimistic upper bound to the optimal performance in every problem instance tested. On average, the expected performance is 98.5% or higher for the different problem sizes. Note that the average performance improves with increasing problem size, as statistical mixing across tasks increases the accuracy of the approximate optimization problem used by the MPC algorithm.

2.5 Continuing Directions of Investigation

The majority of our effort was focused on the allocation of nonrenewable resources such as munitions. Our results provide a design theory for real-time mission control algorithms that determine sequencing of targeting activities over temporal periods, taking into account uncertainty in success of munitions. However, the above models do not address the use of UAVs for surveillance activities, where the expended resources are renewable, such as sensing resources.

In the last year of our research effort, we studied a different class of models: Strategy Development for sensor assignment, based on generalization of surveillance problems involving tactical UAVs such as Predators or smaller UAVs. In these problems, UAV sensors search for an unknown set of mobile, non-cooperative targets, and attempt to detect, track and classify these targets. Our focus was on problems where tasks arrive dynamically using stochastic models, and where there is significant risk incurred by the individual vehicles. We developed a mathematical representation of these problems that capture elements of both risk to UAVs, and uncertainty in future arrivals of tasks. We developed control algorithms for such problems based on receding horizon planning using a mixed integer linear programming formulation, as well as alternative approaches using Lagrangian

relaxation and rollout algorithms [20]. Initial results were presented in [11]. The results of this research form the core of Major Darryl Ahner's doctoral dissertation at Boston University, which will be completed May 2005.

2.6 Distributed Algorithms

The previous discussion focused on mission control algorithms that would be executed at a central location, such as a ground-control station, and thus assume centralization of information. In order to address issues of distributed mission control, where multiple agents have access to different information, we studied a class of distributed task assignment problems where multiple UAVs cooperate to solve optimization problems using only local, possibly outdated information. Specifically, we studied the problem of task partitioning among members of a team of cooperating agents, each with potentially different local information. In these problems, each agent has finite resources, and has an initial set of tasks assigned to it that represents an optimal partitioning given the initial centrally known information. The problem of interest is stated as follows: Given a finite sequence of new task arrivals, observed locally by different agents at different times, find an optimal redistribution of the old and new tasks among the agents in the presence of different information.

One possible approach to this problem is to wait until all of the local information is centrally acknowledged by all agents. At this time, every agent could execute an optimal reassignment algorithm to compute the new task assignments. However, this approach can incur significant delays waiting to obtain synchronization of information. An alternative is to allow each agent to compute modifications to the current ask assignments without waiting for information synchronization. Each agent computes a *speculative* adjustment to assignments in response to each new task it detects, and communicates these adjustments to other team members.

In [9], we explored this approach, and developed a distributed, asynchronous reassignment algorithm that responds to new events locally without waiting for information synchronization. We extended previous results in asynchronous algorithms for parallel solution of assignment problems [3, 4, 2] that addressed problems with known tasks to the case where tasks are dynamically arriving and information differences occur among agents. Specifically, we developed a new class of asynchronous task algorithms that generalize the distributed sequential shortest path algorithms of [4, 2]. In our algorithms, agents respond to the discovery of new tasks by suggesting redistribution of tasks among agents. Since agents observe arrivals of the same task at different times, our algorithm in

[9] often encounters multiple agents reacting to the same task, and includes a protocol for conflict resolution as well as information communication among agents.

Our main result shows that, as long as communication delays are bounded, the distributed assignment algorithms converge to an optimal assignment once all the tasks have arrived. Furthermore, the distributed algorithms are constantly shifting responsibility for tasks locally between agents in response to new task arrivals that are detected only locally, instead of waiting for all information to be distributed globally. The convergence results depend critically on the presence of a *validation* agent, whose role it is to reject assignment modifications suggested by agents when the information used to generate such modifications is deemed to be too far out-of-date. Our algorithms describe a validation protocol that guarantees that a complementary slackness condition is maintained by all accepted changes. This invariance is exploited to prove the convergence to optimal reassignments once new task arrivals stop.

The distributed algorithms have a significant advantage over centralized algorithms: They respond promptly to detection of new events, without delays required to synchronize information with other agents. To illustrate the performance of the distributed algorithm, we ran a simple experiment with 10 agents, each carrying 10 resources. The vehicles were evenly distributed in a square grid. There were up to 100 task arrivals, randomly distributed with a uniform distribution on the square grid. Each task arrival was observed only by the closest agent to it. The value of each task assignment to each agent was inversely proportional to the distance from each agent, plus a random component with standard deviation equal to 10% of the maximum value.

When an agent detected a task arrival, it computed desired modifications to task assignments and waited for a turn to communicate with the validation agent. We simulated a cyclic slotted communication system, where each agent can communicate with the validation agent once a cycle, which occurs after a fixed number of external task arrivals. The agents communicate in fixed order; agents that communicate first are unaware of information provided by agents that communicate later. This creates asymmetries in information due to two effects: between communication slots, new task arrivals are known only to their closest agents, and during communication times, agents only know the information provided to the validation agent by other agents that communicated earlier in the cycle.

Table 4 shows the averaged results of Monte Carlo experiments respect to task arrivals. The results

	Total Number of Tasks					
Arrivals/cycle	75	80	85	90	95	100
2	0	0	0.01	0.01	0.015	0.02
3	0	0	0.01	0.015	0.022	0.028
5	0	0	0.012	0.015	0.028	0.037
10	0	0.005	0.016	0.03	0.035	0.055

Table 4: Fraction of incompatible assignment modifications vs. delays

show that, even when we have significant delays in information synchronization, the average number of tentative assignment modifications (augmentations) that were rejected was minimal, indicating that our speculative computation approach can provide significant advantages over a centralized approach.

3 Dynamic Vehicle Assignment and Scheduling

In the previous section, we summarized our results on Strategy Development, looking at algorithms that compute approaches for developing task strategies and allocation without focusing on detailed trajectory control and scheduling. In order to generate actual control trajectories, lower level algorithms are needed to compute actual trajectories and schedules of activities. In this section, we summarize our results on Dynamic Vehicle Assignment and Scheduling.

We consider multiple vehicles headed for multiple target points to collect rewards associated with them. The team objective is to maximize the total reward accumulated over a given time interval. Complicating factors include uncertainties regarding the location of target points and the effectiveness of collecting rewards, heterogeneous vehicle capabilities and time-varying rewards. The problem consists of assigning tasks and arrival times for each vehicle, and selecting trajectories from each vehicle to its assigned task. A motivating paradigm for this problem is the assignment of smart autonomous munitions to a group of targets.

The mission control algorithms discussed in Strategy Development work with a highly-aggregated representation of the continuous trajectories that vehicles must fly. These aggregate problems serve to identify the tasks that should be scheduled for individual vehicles by the lower level dynamic vehicle assignment and scheduling. The problem of interest is controlling the movement of the UAVs and ultimately assigning them to targets so as to maximize the total reward collected by visiting targets within a given mission time. The problem is complicated by several factors: (i) Target rewards may be time-dependent; (ii) Different vehicles have different capabilities in terms of collecting the reward associated with a target; (iii) The exact location of targets may not always be known; (iv) There may be obstacles which constrain the feasible trajectories of vehicles or may cause their elimination when they are encountered, (v) Information about the state of the mission space is incomplete and dynamic.

The above problems are complex, and are often approached using a functional decomposition that solves task assignment separately from coarse routing and fine routing. We developed an alternative based on temporal decomposition, solving an optimization problem with a receding horizon. An advantage of this approach is that it integrates the three tasks of UAV assignment to targets, routing of UAVs to their assigned targets, and real-time trajectory generation, all in one function. The main idea of the proposed control scheme is to dynamically determine vehicle trajectories by

solving a sequence of optimization problems over a planning horizon and executing them over a shorter action horizon. Thus, we replace a complex discrete stochastic optimization problem by a sequence of simpler continuous optimization problems.

3.1 Formulation

In our research, we considered a setting which involves a team of N UAVs indexed by $j = 1, \ldots, N$ and a set of M *targets* indexed by $i = 1, \ldots, M$ in a 2-dimensional space. Associated with the ith target is a reward R_i. A *mission* is defined as the process of controlling the movement of the UAVs and ultimately assigning them to targets so as to maximize the total reward collected by visiting targets within a given mission time T.

In a 2-dimensional mission space, the location of the ith target is denoted by $y_i \in \mathbb{R}^2$, $i = 1, \ldots, N$, and the position of the jth vehicle at time t is denoted by $x_j(t) \in \mathbb{R}^2$, $j = 1, \ldots, M$. The vehicles' intial positions are given by x_{j0}, $j = 1, \ldots, M$. Assuming a vehicle travels at constant velocity throughout the team mission, the vehicle dynamics are

$$\dot{x}_j(t) = V_j \begin{bmatrix} \cos u_j(t) \\ \sin u_j(t) \end{bmatrix}, \quad x_j(0) = x_{j0}$$

where $u_j(t) \in [0, 2\pi]$ is the heading (the control variable) of vehicle j and V_j is the corresponding velocity.

Vehicles complete tasks in the mission space by visiting targets. To distinguish the importance of tasks at time t, each target has an associated *reward function* denoted by $R_i\phi_i(t)$, where R_i is the maximal reward and $\phi_i(t) \in [0, 1]$ is a discounting function which describes the changing of reward with time. This allows us to model targets that must be visited by some deadline or, more generally, targets whose rewards are positive only within some given time window. To distinguish between the effectiveness of vehicles relative to a target i, we define a *capability factor* $p_{ij}(t) \in [0, 1]$, which reflects the probability that a UAV j visiting target i at time t will complete the task and collect the reward $R_i\phi_i(t)$. The objective of the mission is to collect the maximal total reward by the end of some mission time T.

3.2 Receding Horizon Control

To meet this goal, we developed a cooperative controller updated at time points denoted by t_k, $k = 0, 1, \ldots$, during the mission time. At t_k, the controller operates by solving an optimization problem $\mathbf{P}_k$, whose solution is the control vector $\mathbf{u}_k = [u_1(t_k) \ldots u_M(t_k)]$. We explain next how $\mathbf{P}_k$ is formulated (see also [6]).

Suppose that vehicles are assigned headings $u_1(t_k), \ldots, u_M(t_k)$ at time t_k and maintain them for a *planning horizon* denoted by H_k. Then, at time $t_k + H_k$ the positions of the vehicles are given by

$$x_j(t_k + H_k) = x_j(t_k) + \dot{x}_j(t_k)H_k$$

Define

$$\tau_{ij}(t_k, \mathbf{u}_k) = (t_k + H_k) + ||x_j(t_k + H_k) - y_i||/V_j \tag{8}$$

and note that $\tau_{ij}(t_k, \mathbf{u}_k)$ is the earliest time that vehicle j can reach target i under the condition that it starts at t_k with control dictated by $\mathbf{u}_k$ and then proceeds directly from $x_j(t_k + H_k)$ to the target point y_i ($||\cdot||$ is the usual Euclidean norm). We are interested in the maximal reward that vehicle j can extract from target i if it reaches the target at time $\tau_{ij}(t_k, \mathbf{u}_k)$. Clearly, this is given by $R_i\phi_i[\tau_{ij}(t_k, \mathbf{u}_k)]$. For convenience, define

$$\tilde{\phi}_{ij}(t_k) = \phi_i[\tau_{ij}(t_k, \mathbf{u}_k)] \tag{9}$$

where it is worth pointing out that $\tilde{\phi}_{ij}(\cdot)$, unlike $\phi_i(\cdot)$, depends on both i and j. It is also clear that the probabilty of extracting this reward, evaluated at time t_k, is given by $p_{ij}[\tau_{ij}(t_k, \mathbf{u}_k)]$. For convenience, we set

$$\tilde{p}_{ij}(t_k) = p_{ij}[\tau_{ij}(t_k, \mathbf{u}_k)] \tag{10}$$

Next, let us define the *relative distance* function, $\delta_{ij}(t)$, defined as

$$\delta_{ij}(t) = \frac{||x_j(t) - y_i||}{\sum_{k=1}^{M} ||x_k(t) - y_i||} \tag{11}$$

which provides the proximity of vehicle j to target point i relative to other vehicles, given all vehicle positions $x_1(t), \ldots, x_M(t)$. Based on this, let us define a *normalized relative distance function* $q_{ij}(\delta_{ij})$ to be a monotonically nonincreasing function such that

$$q_{ij}(0) = 1, \quad q_{ij}(1/M) = 1/M, \quad \lim_{\delta_{ij} \to 1} q_{ij}(\delta_{ij}) = 0 \tag{12}$$

which we view as the probability that target i is assigned to vehicle j at time t (thus, when $\delta_{ij} = 0$ for example, vehicle j is "committed" to target i). We are interested in the value of this function at $t = t_k + H_k$ and define

$$\tilde{q}_{ij}(t_k) = q_{ij}[\delta_{ij}(t_k + H_k)] \tag{13}$$

We can now present the optimization problem $\mathbf{P}_k$, formulated at time t_k, as follows:

$$\max_{\mathbf{u}_k} \sum_{i=1}^{N} \sum_{j=1}^{M} R_i \tilde{\phi}_{ij}(t_k) \tilde{p}_{ij}(t_k) \tilde{q}_{ij}(t_k) \tag{14}$$

with $\tilde{\phi}_{ij}(t_k)$, $\tilde{p}_{ij}(t_k)$, and $\tilde{q}_{ij}(t_k)$ as defined in (9), (10), and (13) respectively. The expression $R_i \tilde{\phi}_{ij}(t_k) \tilde{p}_{ij}(t_k) \tilde{q}_{ij}(t_k)$ in (14) can be seen as the expected reward that vehicle j collects at target i, evaluated at time t_k using a planning horizon H_k. Similar to $p_{ij}(t)$, we can also define $r_{ij}(t) \in [0,1]$ to be the ith target's capability factor when inflicting damage to vehicle j. Letting C_j be the cost of losing the jth vehicle, we can extend (14) above to include expected costs due to UAV loss:

$$\max_{\mathbf{u}_k} \left[\sum_{i=1}^{N} \sum_{j=1}^{M} R_i \tilde{\phi}_{ij}(t_k) \tilde{p}_{ij}(t_k) \tilde{q}_{ij}(t_k) - \sum_{i=1}^{N} \sum_{j=1}^{M} C_j \tilde{r}_{ij}(t_k) \right] \tag{15}$$

where $\tilde{r}_{ij}(t_k)$ is defined similar to $\tilde{p}_{ij}(t_k)$. Note that this is a nonlinear optimization problem whose solution is generally easy to obtain compared to a combinatorial optimization problem seeking to assign UAVs to targets over the entire mission time.

Upon getting the optimal $\mathbf{u}_k$ for (15) based on all state information available at t_k, the UAV team will follow this control for an *action horizon* $h_k \leq H_k$. The process is then repeated at time

$$t_{k+1} = t_k + h_k, \quad k = 0, 1, \ldots$$

The value of h_k is determined by two factors. First, if an unexpected event takes place at some $t_e \in (t_k, t_k + h_k)$, then we set $h_k = t_e - t_k$. Otherwise, we simply update the control after the prespecified amount of time h_k. Thus, in general, $\{t_k\}$ is a random sequence.

The choices of the functions $\phi_i(t)$ and $q_{ij}(\delta_{ij})$ may depend on a variety of factors, including the possible presence of hard deadlines on visiting target points and the desirability of maintaining tight vehicle formations as opposed to spreading the vehicles out over the full mission space. Some such choices were given in [6].

3.3 Results

The properties of the cooperative controller outlined above were tested in a variety of simulated environments and found to match a reward upper bound (obtained by an exhaustive search of all possible vehicle-to-target assignments and minimal straight line trajectories) with high probability. The most surprising finding of this work to date [8], however, has been the fact that vehicles are *always ultimately assigned to target points*, despite the fact that our approach, by its nature, was never intended to actually perform any such assignment. A salient property of this scheme, which selects real-valued vehicle headings, is to enforce the convergence of these headings to values that reflect discrete vehicle-to-target point assignments. This intriguing observation has broad implications in tackling such hard stochastic and intrinsically discrete optimization problems.

Looking at (15), note that the solution of this problem is a real-valued heading $u_j(t_k)$ for every vehicle $j = 1, \dots, M$. There is no express constraint imposed to assign a vehicle to a target location, i.e., there is no constraint of the form $x_j(t_k + H_k) \in \{y_1, \dots, y_N\}$ or $\frac{y_i - x_j(t_k)}{\|y_i - x_j(t_k)\|} = \frac{x_j(t_k+H_k) - x_j(t_k)}{\|x_j(t_k+H_k) - x_j(t_k)\|}$ forcing a vehicle to either be at a target by a certain time or to set a heading for it. The intent of designing the RH controller was originally to simply direct vehicles toward points in the mission space in a way that the team maximizes an *expected* reward as represented by the objective function in (15); a lower-level controller might then be responsible for making final *discrete* assignments. However, the empirical observation made was that in fact each vehicle always eventually ends up visiting a target point, as long as the planning horizon is selected to be $H_k = \min_{j \in \mathcal{A}, i \in \mathcal{T}} \{\|y_i - x_j(t_k)\| / V_j\}$, the smallest distance (in time units) between any vehicle and any target point at time t_k.

In order to prove this convergence property, we used the theory of potential functions. Intuitively, we may view a target reward as the value of a potential function associated with that target; this potential then exerts an "attractive force" on UAVs that "compete" for assignment to the target. Using this framework, we introduce the concept of a *stationary* UAV trajectory in the sense that *it always ends up assigned to a specific target* (as opposed to a point in space from which it would be the responsibility of a separate controller to assign it to a target). By analyzing the dynamics of a simple setting in [8], we have obtained convergence proofs for simple vehicle scenarios under simple restrictions on the relative values of the tasks.

Our discussion above has been based on the assumption that all target locations are originally known and given by y_i, $i = 1, \dots, N$. Clearly, it is desirable to endow a UAV team with the versatility

to adjust its cooperative behavior as new targets are detected. The very nature of the RH control approach makes it amenable to this capability: since the controller is updated whenever a new event occurs, we can simply define the detection of a new target as such a "new event". Consequently, the presence of a new target causes changes in (15) when it is solved following this event, but there is no fundamental difficulty expected other than an increase in the dimensionality of the problem since there are now $N+1$ targets rather than N.

A phenomenon that we observed in our work is that of occasional oscillatory behavior in the trajectories of certain UAVs. This is illustrated in Fig. 1 (based on simulation experiments), where the RH controller is used with two UAVs and six targets. The UAV on the right side immediately experiences oscillations and the reason is that it happens to be initially positioned roughly equally close to targets 5 and 6 which have comparable characteristics. A similar situation arises for the UAV on the left side when it reaches the vicinity of targets 1 and 4. Although this behavior is seen to be ultimately resolved, it is clearly undesirable. The root of this problem lies in the formulation of (15) which assumes that a vehicle is free to arbitrarily change directions at no cost.

We eliminated this behavior by introducing a cost (i.e., a negative reward in (15)) to changing directions. The direction change cost function is $\Delta(u, u')$, where u is the current heading of a vehicle and u' is a new heading, determined as the solution of a problem of the form $\mathbf{P}_k$ at some t_k. Using this approach, the oscillatory behavior of of Fig. 1 is eliminated.

3.4 Continuing research direction

There are several issues that remain for future investigation. One of them is the choice of functions $\phi_i(t)$ and $q_{ij}(\delta_{ij})$ (and hence $\tilde{\phi}_{ij}(t_k)$ and $\tilde{q}_{ij}(t_k)$) depends on several factors. However, the basic structure of these functions is well-defined, given requirements such as (12). It is reasonable to conjecture that the effect of altering the "shape" of the functions is not significant on the behavior of the RH controller. This is an issue that deserves further investigation.

Another important extension is to deal with stand-off capabilities on UAVs. Often, a UAV can perform a task at a significant stand-off distance. Platforms such as Global Hawk have radars capable of imaging objects 100 km away from the platform. An abstraction of this problem is to have UAVs reach a distance to a task in an interval $[d_{\min}, d_{\max}]$ in order to be able to perform this task.

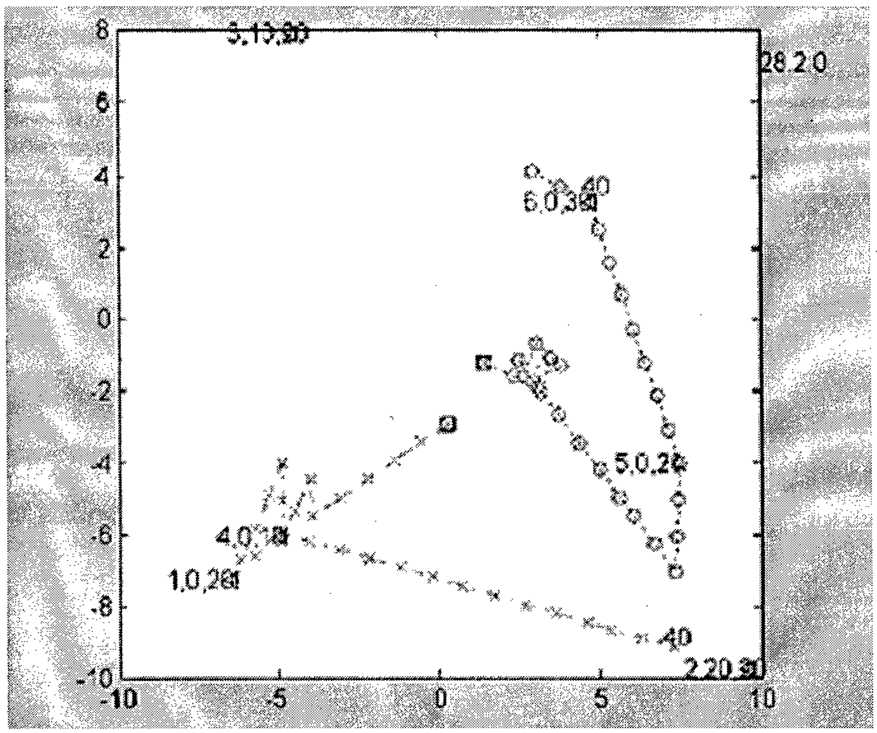

Figure 1: Example of target assignment instability

A third extension is dealing with search activities. Our discussion above has been based on the assumption that all target/task locations are known and given by y_i, $i = 1, \ldots, N$. Clearly, it is desirable to endow a UAV team with the versatility to search for new targets. Although in principle we believe the same approach can be adopted for such tasks, this is a research direction that remains for future investigations.

4 Publications

The following publications document in greater detail the work discussed above.

1. "A Receding Horizon Approach for Solving Some Cooperative Control Problems," C. G. Cassandras and W. Li, *Proc. 2002 Conference on Decision and Control,* Las Vegas, Nevada, December 2002.

2. "Model Predictive Control for Dynamic Unreliable Resource Allocation," D. A. Castañón and J. M. Wohletz, *Proc. 2002 Conference on Decision and Control,* Las Vegas, Nevada, December 2002.

3. "Model Predictive Control for Stochastic Dynamic Resource Allocation," D. A. Castañón, submitted to *IEEE Transactions on Automatic Control.*

4. "Distributed Algorithms for Reassignment," D. A. Castañón and C. Wu, *Proc. 2003 Conference on Decision and Control,* Maui, HI, December 2003.

5. " Stability Properties of a Cooperative Receding Horizon Controller," C. G. Cassandras and L. Wei, *Proc. 2003 Conference on Decision and Control,* Maui, HI, December 2003.

6. "Stability Properties of a Receding Horizon Contoller for Cooperating UAVs," W. Li and C. G. Cassandras, to appear in *Proc. of 43rd IEEE Conf. Decision and Control,* 2004.

7. "A Cooperative Receding Horizon Controller for Multi-Vehicle Uncertain Environments," Li, W., and Cassandras, C.G., subm. to *IEEE Trans. on Automatic Control,* 2004.

8. "Planning and Control of UAVs as a Dynamic and Stochastic System ," D. Ahner and D. A. Castañón, presented at MORS 2004, Monterey, CA.

References

[1] Chairman of the Joint Chiefs of Staff, "Joint vision 2010," 1999.

[2] D. P. Bertsekas and D. A. Castañón, "Parallel Synchronous and Asynchronous Implementations of the Auction Algorithm," *Parallel Computing*, V 17, Sept. 1991, pp. 707–732.

[3] D. P. Bertsekas and D. A. Castañón, "Parallel Asynchronous Hungarian Methods for the Assignment Problem," *ORSA Journal on Computing*, V 5, No. 3, Summer, 1993.

[4] D. P. Bertsekas and D. A. Castañón, "Parallel Primal-Dual Methods for the Minimum Cost Network Flow Problem," *Computational Optimization and Applications*, V 2, pp. 319–338, 1993.

[5] D. A. Castañón and J. M. Wohletz, "Model Predictive Control for Dynamic Unreliable Resource Allocation," in *Proc. of 41st IEEE Conf. on Decision and Control*, pp. 3574–3579, 2002.

[6] C. G. Cassandras and W. Li, "A Receding Horizon Approach for Solving Some Cooperative Control Problems," in *Proc. of 41st IEEE Conf. on Decision and Control*, pp. 3760–3765, 2002.

[7] R. Bachmayer and N. E. Leonard, "Vehicle Networks for Gradient Descent in a Sampled Environment," in *Proc. of 41st IEEE Conf. on Decision and Control*, pp. 112–117, 2002.

[8] W. Li and C. G. Cassandras, "Stability Properties of a Cooperative Receding Horizon Controller," *42nd IEEE Conf. on Decision and Control*, Maui, HI, 2003.

[9] D. A. Castañón and C. Wu, "Distributed Algorithms for Dynamic Reassigment," in *42nd IEEE Conf. on Decision and Control*, Maui, HI, 2003.

[10] D. A. Castañón and J. M. Wohletz, "Model Predictive Control for Stochastic Dynamic Resource Allocation," submitted to *IEEE Transactions on Automatic Control.*

[11] D. Ahner and D. A. Castañón," Planning and Control of UAVs as a Dynamic and Stochastic System ," Presented at MORS 2004, Monterey, CA.

[12] J. S. Bellingham, M. Tillerson, M. Alighanbari, and J. P. How, "Cooperative Path Planning for Multiple UAVs in Dynamic and Uncertain Environments," in *Proc. of 41st IEEE Conf. on Decision and Control*, pp. 2816–2822, 2002.

[13] J.-C. Latombe, *Robot Motion Planning.* Kluwer Academic Publishers, 1991.

[14] C. G. Cassandras, *Discrete Event Systems: Modeling and Performance Analysis.* Homewood, IL: Irwin Publ., 1993.

[15] K. Gokbayrak and C. G. Cassandras, "Hybrid controllers for hierarchically decomposed systems," in *Proceedings of 3rd Intl. Workshop on Hybrid Systems: Computation and Control*, pp. 117–129, March 2000.

[16] K. Gokbayrak and C. G. Cassandras, "A hierarchical decomposition method for optimal control of hybrid systems," in *Proceedings of 39th IEEE Conf. On Decision and Control*, pp. 1816–1821, December 2000.

[17] M. Pachter and P. R. Chandler, "Challenges of autonomous control," *IEEE Control Systems Magazine*, August 1998.

[18] P. R. Chandler, M. Pachter, and S. Rasmussen, "UAV Cooperative Control," in *Proc. of 2001 American Control Conference*, pp. 50–55, 2001.

[19] B. T. Clough, "Unmanned Aerial Vehicles: Autonomous Control Challenges, a Researcher's Perspective," in *Cooperative Control and Optimization* (R. Murphey and P. M. Pardalos, eds.), pp. 35–53, Kluwer Academic Publishers, 2000.

[20] D. P. Bertsekas and D. A. Castañón, " Rollout Algorithms for Stochastic Scheduling ," *Heuristics*, April 1999.

[21] J. M. Wohletz, D. A. Castañón, and M. L. Curry, "Closed-Loop Control for Joint Air Operations," in *Proc. of 2001 American Control Conference*, pp. 4699–4704, 2001.